EasyTrage

Team Trage:

Mike Hamilton (initial Scrum Master),
Kevin Lee,
Abdul Mohamed
Vikram Melkote (Product Owner)
Christopher Kong

Motivation

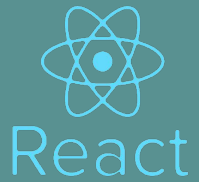
- In the Crypto world there are a lot of different cryptocurrency exchanges that allow users to trade different digital assets for their native currency as well as other digital assets.

- Whenever the prices of one or more cryptocurrencies plunges or has a run up, there is often a lag in price movement between different exchanges, especially ones hosted in different countries. We want to build a bot that capitalizes on this.

High Level Goals

- Create a website that is capable of trading cryptocurrencies/suggesting trades through different exchanges
- Has the ability to compare prices, fees, transfer times of cryptocurrencies on different exchanges to find out which trades are profitable.
- Displays the discrepancies between exchanges
- Login and sign-up functionality with history data for trades through the api.

Technologies



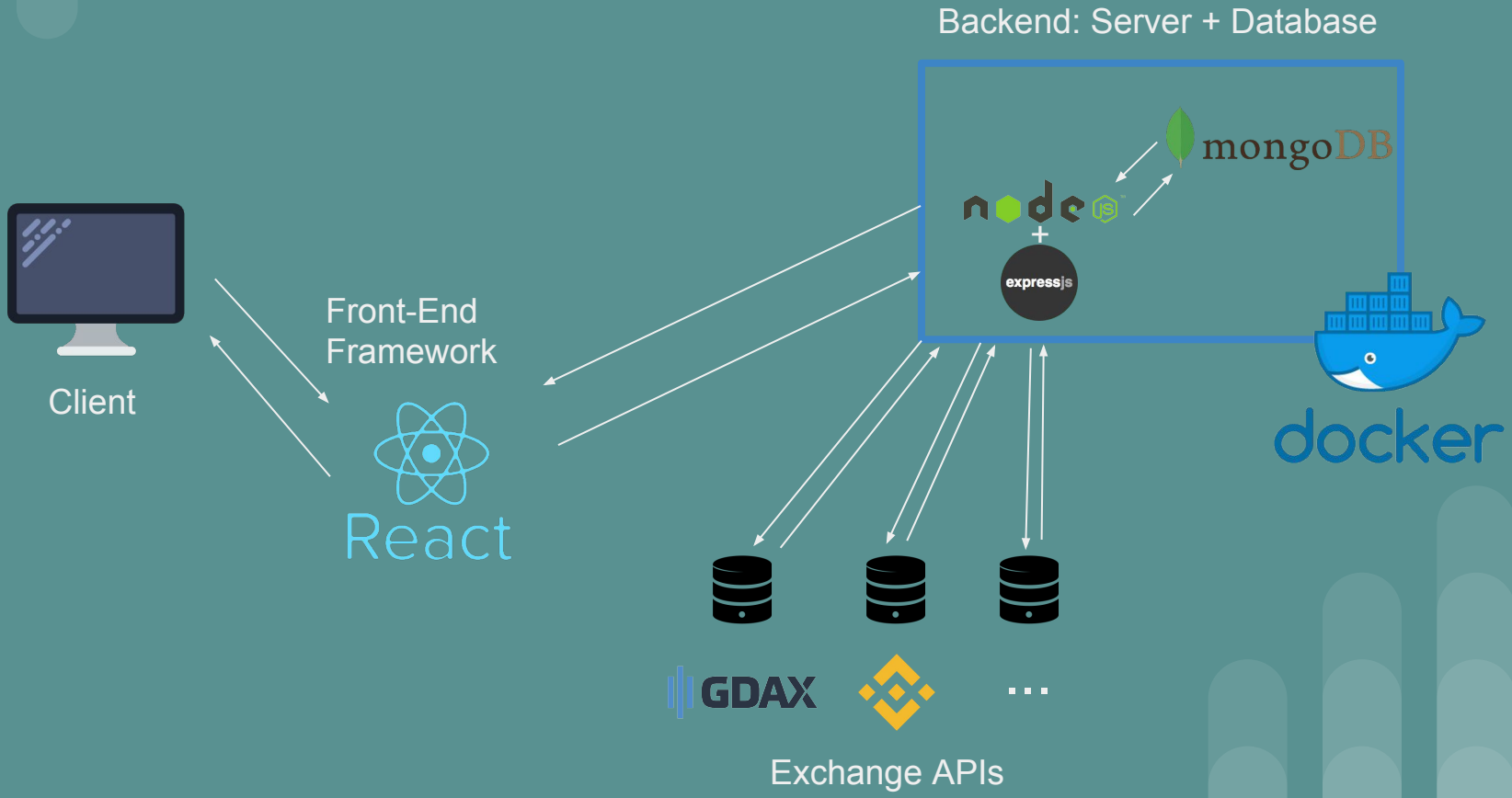
React - Front end framework to handle templating/ HTML generation, displaying data from web server / db for user

mongoDB - database for storing usernames, emails, hashed passwords, persistent site data

Node.js + Express.js - Back end / server side frameworks for interacting with other API servers of exchanges, checking prices, performing calculations routinely to look for trades, etc

Docker - containerizing developer tool so that we don’t have the “it works on mine” problem, and if we want to deploy it, the deployment environment will be the same as the development environment

Architecture



Sprint 1

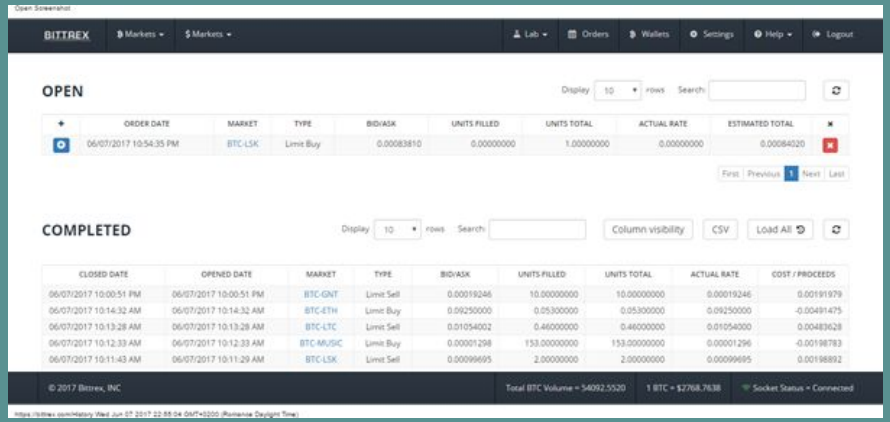
- (5 story points) User Story 1: As developers, we want to familiarize ourselves with React and go through some basic tutorials to learn the syntax and architecture
- (3 story points) User story 2: As an uninitiated user, I want to be able to successfully create an account on EasyTrage and have my data persist in a database
- (3 story points) User story 3: As a potential investor, I want the site to look attractive and be easily readable with good design

Sprint 2

- (5 story points) User story 1: As a cryptocurrency trader, I want to be able to view current coin prices (in native currency) from various exchanges
- (5 story points) User story 2: As an EasyTrage user, I want to see trade fees, current estimated transaction time, estimated transaction fees for coins from various exchanges
- (8 story points) User story 3: As an EasyTrage user, I want to see trade suggestions for at least 1 of coin

Sprint 3

- (13 story points) User story 1: As a user, I want to be able to input my API keys of my exchange accounts and have the bot trade on my behalf (if trade suggestions are satisfactorily approved by the team at this point)
- (5 story points) User story 2: As a user, I want to see all my past trades and profits/losses



Sprint 4

- (5 points) User story 1: As a developer, I want to polish up the interface and make the site visually appealing to my users
- (13 points) User story 2: As a user, I want more than 1 option for coins the webapp can suggest/trade to increase the number of possible trades I can profit from

Backlog

- (5 points) As a user, I want to have 2-Factor Authentication via google authenticator for security so malicious parties cannot access my account without my phone
- (8 points) As a user, I want to see a visualization of differences between prices for at least 1 coin
- (8 points) As a user, I want to see a visualization for my profit/loss over time

Challenges and Risks

- APIs have call rate limits and may not have the most recent information which may make reported trades harmful for the user
- Some coins take a long time to transfer and the user might be stuck waiting for a transfer and then forced to take a loss instead of making a profit
- Trading and transaction fees may mitigate some profits entirely or even turn them into losses, so we must try to factor them in